Comp. & Maths. with Appls., Vol. 5, pp. 211–216

0097-4943/79/0901-0211/$02.00/0

A GENERALISED SPARSE FACTORISATION METHOD FOR THE SOLUTION OF PERIODIC TRIDIAGONAL SYSTEMS

D. J. EVANS and S. O. OKOLIE
Department of Computer Studies, Loughborough University of Technology, Loughborough, Leicestershire, England

Communicated by E. Y. Rodin

(*Received February* 1979)

Abstract—Previously proposed factorisation methods for the solution of the general periodic tridiagonal matrix systems have not been entirely successful in maintaining the sparsity of the given coefficient matrix in the factorisation process. Here, a general sparse cyclic factorisation method involving a continued fraction expansion whilst maintaining both the sparsity and form of the given coefficient matrix is proposed.

The resulting algorithm solution is shown to be a very fast method which is recommended for use in problems requiring a repeated solution of the matrix system as in the solution of the Crank–Nicolson finite difference equations derived from linear self-adjoint parabolic problems under periodic spatial boundary conditions.

1. INTRODUCTION

In the numerical solution of many physical and engineering problems, such as the general diffusion equation with periodic spatial conditions, Evans[1], and in spline approximations[2] when a non-uniform mesh discretisation is introduced, there occurs the need to solve repeatedly systems of linear equations of the form

$$\mathbf{Au} = \mathbf{d} \tag{1.1}$$

where A is a general periodic tridiagonal matrix,

$$A = \begin{bmatrix} b_1 & c_1 & & & a_1 \\ a_2 & b_2 & c_2 & & 0 \\ & \ddots & \ddots & \ddots & \\ 0 & & \ddots & \ddots & c_{n-1} \\ c_n & & & a_n & b_n \end{bmatrix}_{(n\times n)} \tag{1.2}$$

which would normally be positive definite and diagonally dominant.

Available factorisation methods for the solution of (1.1) have not been entirely successful in maintaining the matrix sparsity and form in the factorisation process[3] and those that maintain sparsity (and hence provide exact techniques) are only applicable in special cases where A is both symmetric and circulant[1]. In this paper, a general sparse cyclic factorisation method which maintains both the form and sparsity of the coefficient matrix is presented for the solution of (1.1) and shown to provide a very fast method in applications where a repeated solution of the matrix system is required (with the coefficient matrix remaining unchanged) as is the case in the solution of linear self-adjoint parabolic partial differential equations.

2. DERIVATION OF ALGORITHMIC SOLUTION

For the solution of the linear equation (1.1), we propose a general sparse cyclic factorisation of the matrix A into a lower cyclic matrix P and an upper cyclic matrix Q where both matrices have the form:

$$P = \begin{bmatrix} 1 & & & l_1 \\ l_2 & 1 & 0 & \\ & \ddots & \ddots & \\ 0 & & \ddots & 1 \\ & & l_n & \end{bmatrix} \quad \text{and} \quad Q = \begin{bmatrix} v_1 & c_1 & & \\ & v_2 & c_2 & 0 \\ & & \ddots & \ddots \\ 0 & & \ddots & c_{n-1} \\ c_n & & & v_n \end{bmatrix}. \tag{2.1}$$

By forming the product PQ and equating the elements to the corresponding elements of A, a set of equations which yield the elements of P and Q is derived to give,

$$\left.\begin{aligned} &l_i = a_i/v_{i-1}, \\ \text{and} \quad &v_i = b_i - l_i c_{i-1}, \\ \text{with} \quad &c_0 \equiv c_n, \quad v_0 \equiv v_n. \end{aligned}\right\} \qquad i = 1, 2, \ldots, n-1. \tag{2.2}$$

The merit of the above factorisation is that both the form and sparsity of the original matrix are preserved. However, the elements l_i, v_i in (2.2) cannot be determined uniquely unless one of the elements, typically l_1, is obtained first in some efficient manner. We shall adopt the continued fraction approach in this regard (see Section 3) and assume for the moment that l_1 has been uniquely determined. It then becomes easy to compute the remaining terms l_i, v_i using (2.2) in order to complete the factorisation.

For the solution of (1.1) we can therefore consider the two alternative forms,

$$P\mathbf{y} = \mathbf{d} \tag{2.3}$$

and

$$Q\mathbf{u} = \mathbf{y} \tag{2.4}$$

where $\mathbf{y}$ is introduced as an auxiliary vector.

First, we consider the system (2.3). To solve for $\mathbf{y}$ we multiply the 1st equation of the system by $-l_2$ and add the result to the 2nd equation to obtain a new second equation. Next, we multiply the new 2nd equation by $-l_3$ and add the result to the 3rd equation to obtain a new 3rd equation, etc.

This process is continued as far as the nth equation, resulting in the following reduced system,

$$\begin{bmatrix} 1 & & & \phi_1 \\ & 1 & & \phi_2 \\ & & \ddots & \vdots \\ & 0 & & 1 \quad \phi_{n-1} \\ & & & 1+\phi_n \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} d_1' \\ d_2' \\ \vdots \\ d_n' \end{bmatrix} \tag{2.5}$$

where

$$\phi_i = (-1)^{i-1} \prod_{j=1}^{i} l_j, \qquad i = 1, 2, \ldots, n \tag{2.6}$$

$$d_i' = d_i - l_i d_{i-1}', \qquad i = 1, 2, \ldots, n \tag{2.7}$$

with $d_0' \equiv 0$.

From (2.5), y_i is easily obtained as

$$\left.\begin{aligned} &y_n = d_n'/(1+\phi_n) \\ \text{and} \quad &y_i = d_i - \phi_i y_n, \end{aligned}\right\} \qquad i = n-1, n-2, \ldots, 1. \tag{2.8}$$

Next we consider the system (2.4) and rewrite it in the normalised matrix form,

$$\begin{bmatrix} 1 & \theta_1 & & \\ & 1 & \theta_2 & 0 \\ & & \ddots & \theta_{n-1} \\ & 0 & & 1 \\ \theta_n & & & \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix} = \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_n \end{bmatrix} \tag{2.9}$$

where, for $i = 1, 2, \ldots, n$

$$\theta_i = c_i/v_i \tag{2.10}\dagger\dagger$$

and

$$g_i = y_i/v_i. \tag{2.11}**$$

To solve the system (2.9) we multiply the nth equation by $-\theta_{n-1}$ and add to the $(n-1)$th equation to obtain a new $(n-1)$th equation. This process is continued until finally, we multiply the new second equation by $-\theta_1$ and add this to the 1st equation to obtain a new 1st equation. The resulting system becomes

$$\begin{bmatrix} 1+\gamma_1 & & & \\ \gamma_2 & 1 & & 0 \\ \vdots & & \ddots & \\ \gamma_n & 0 & & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix} = \begin{bmatrix} g_1' \\ g_2' \\ \vdots \\ g_n' \end{bmatrix} \tag{2.12}$$

where

$$\left.\begin{aligned} \gamma_i &= (-1)^{i-1} \prod_{j=i}^{n} \theta_j, \quad i = 1, 2, \ldots, n, (n \text{ odd}) \\ &= (-1)^{i} \prod_{j=i}^{n} \theta_j, \quad i = 1, 2, \ldots, n, (n \text{ even}) \end{aligned}\right\} \tag{2.13}\dagger\dagger$$

and

$$\left.\begin{aligned} g_n' &= g_n, \\ g_i' &= g_i - \theta_i g_{i+1}', \end{aligned}\right. \quad i = n-1, n-2, \ldots, 1. \tag{2.14}**$$

Finally, the solution vector is then easily obtained from (2.12) to give,

$$\left.\begin{aligned} u_1 &= g_1'/(1+\gamma_1) \\ u_i &= g_i' - \gamma_i y_1, \end{aligned}\right. \quad i = 2, 3, \ldots, n. \tag{2.15}**$$

In the elimination process outlined above both l_i and θ_i are used as multipliers and hence the stability of the solution process is guaranteed if the absolute values of these multipliers are less than unity. These conditions can be shown to hold under diagonal dominance conditions of the coefficient matrix.

The outlined algorithmic method requires $4n$ multiplications, n divisions and $4n$ additions (as given by equations marked by **) and $5n$ pre-computed coefficients (indicated by ††) provided that at the factorisation stage the term l_1 is determined in the most efficient manner.

For the simple tridiagonal case when $a_1 = c_n = 0$, which is often encountered in problems under the more usual Dirichlet's boundary condition, the above algorithm (2.2)–(2.15) can easily be shown to simplify greatly to the following form.

Define,

$$\begin{aligned} &l_1 = 0, \quad v_1 = b_1, \quad z_1 = 1/v_1 \\ &l_i = a_i z_{i-1}, \quad v_i = b_i - l_i c_{i-1}, \quad z_i = 1/v_i, \quad i = 2, \ldots, n, \end{aligned} \tag{2.16}\dagger$$

then

$$d_i' = d_i - l_i d_{i-1}', \quad i = 1, 2, \ldots, n \tag{2.17}*$$

with

$$d_0' \equiv 0$$

and finally,

$$u_n = d'_n z_n$$
$$u_i = g_i - g_{i+1} c_i z_i, \qquad i = n-1, n-2, \ldots, 1. \tag{2.18}*$$

This requires only $3n$ multiplications, $2n$ additions (marked *) and $3n$ pre-computed coefficients (marked †), and only $3n$ multiplications and $2n$ additions for subsequent solution. This compares favourably with the $3n$ multiplications, $4n$ additions required by the Gaussian elimination method [4].

3. DETERMINATION OF l_1 BY A CONTINUED FRACTION EXPANSION

We can express l_1 as a function of the elements of the matrix A by a cyclic application of the formula (2.2) to give an infinite periodic continued fraction (P.C.F.) of the form,

$$c_n l_1 = \Gamma = \underbrace{\left\{ \cfrac{a_1 c_n}{b_n - \cfrac{a_n c_{n-1}}{b_{n-1} - \ldots}} \right.}_{\text{1st cycle}} \cfrac{a_2 c_1}{b_1 - \cfrac{a_1 c_n}{\underbrace{\left\{ \cfrac{a_n c_{n-1}}{b_n - \cfrac{a_n c_{n-1}}{b_{n-1} - \ldots}} \right.}_{\text{2nd cycle}} \cfrac{a_2 c_1}{b_1 - \cfrac{a_1 c_n}{b_n - \ldots}}}} \tag{3.1}$$

Equation (3.1) can be re-written in a more compact form as,

$$\Gamma = \underbrace{\frac{a_1 c_1}{b_n -} \frac{a_n c_{n-1}}{b_{n-1} -} \ldots \frac{a_2 c_1}{b_1 -}}_{\text{1st cycle}} \underbrace{\frac{a_1 c_n}{b_n -} \frac{a_n c_{n-1}}{b_{n-1} -} \frac{a_2 c_1}{b_1 - \ldots}}_{\text{2nd cycle}} \underbrace{\frac{a_1 c_n}{b_n - \ldots}} \tag{3.2}$$

and for simplicity of exposition as,

$$\Gamma = \underbrace{\frac{\alpha_1}{\beta_1 -} \frac{\alpha_2}{\beta_2 -} \ldots \frac{\alpha_n}{\beta_n -}}_{\text{1st cycle}} \underbrace{\frac{\alpha_1}{\beta_1 -} \ldots \frac{\alpha_n}{\beta_n -}}_{\text{2nd cycle}} \underbrace{\frac{\alpha_1}{\beta_1 - \ldots}} \tag{3.3}$$

where

$$\alpha_i = a_{(n-i+2)} c_{(n-i+1)}, \quad \beta_i = b_{(n-i+1)}$$

with

$$(k) \equiv (k \text{ modulo } n).$$

A P.C.F. of the form in (3.3) is said to be generated by a linear fractional transformation [5],

$$T^n(\omega) = \frac{\alpha_1}{\beta_1 -} \frac{\alpha_2}{\beta_2 - \ldots} \frac{\alpha_n}{\beta_n - \omega} \tag{3.4}$$

where ω is termed the fixed point of the linear transformation. Following Wall [5], the value of the infinite P.C.F. in (3.3) is given by

$$\Gamma = c_n l_1 = \max(\omega_1, \omega_2) \tag{3.5}$$

where ω_1, ω_2 are the two roots of the quadratic equation

$$\omega = \frac{E_{k-1}\omega + E_k}{F_{k-1}\omega + F_k} \tag{3.6}$$

and the coefficients E_r, F_r, $r = 0, 1, \ldots, k \leq n$ are given by the recurrence relation,

$$\left.\begin{aligned} &E_0 = 0, && F_0 = 1, \\ &E_1 = \alpha_1, && F_1 = \beta_1, && r = 2, 3, \ldots, k \leq n. \\ &E_r = \beta_r E_{r-1} - \alpha_r E_{r-2}, && F_r = \beta_r F_{r-1} - \alpha_r F_{r-2}, \end{aligned}\right\} \tag{3.7}$$

The quotient E_r/F_r is defined as the rth approximant of the continued fraction (3.4) and the sequence of approximants $\{E_r/F_r\}$ is said to converge after the kth approximant if

$$\left|\frac{E_k}{F_k} - \frac{E_{k-1}}{F_{k-1}}\right| < \epsilon \tag{3.8}$$

for a sufficiently small truncation error tolerance ϵ ($\simeq 10^{-12}$).

From experimental results it is found that for a strong diagonal dominance of the coefficient matrix, the sequence of approximants of the resulting continued fraction (3.1) form a rapidly convergent sequence; and as a general rule, each additional approximant will yield approximately one further correct decimal place than the previous one. Hence, the convergence criterion in (3.8) would normally be satisfied after the order of k levels of the continued fraction, where k is the maximum number of decimal place accuracy of the computer in use. It is therefore only necessary to evaluate the recurrence relation (3.7) to a level $r = k \leq n$ where 10^{-k} is an accepted truncation error tolerance. Thus, the determination of l_1 by the proposed continued fraction method can be relatively inexpensive, particularly for a large order diagonally dominant system.

4. NUMERICAL RESULTS

The general factorisation algorithmic method given in Section 2, was programmed in Fortran, using single precision arithmetic, for the Loughborough University I.C.L. 1904S computer and used to solve a linear self-adjoint parabolic partial differential equation of the form

$$\frac{\partial U}{\partial t} = \frac{\partial}{\partial x}\left(K(x)\frac{\partial U}{\partial x}\right) + \psi(x, t) \tag{4.1}$$

defined in the semi-definite strip $R = \{0 \leq x \leq l, t \geq 0\}$ where $\psi(x, t)$ is a given function of x and t and $U(x, t)$ is required to satisfy the initial condition of the form,

$$U(x, 0) = f(x) \tag{4.2}$$

and the periodic conditions,

$$U(0, t) = U(l, t)$$

and

$$\frac{\partial U}{\partial x}(0, t) = \frac{\partial U}{\partial x}(l, t). \tag{4.3}$$

On applying the Crank–Nicolson finite difference implicit formula it can be shown[1] that the problem (4.1)–(4.3) leads to a solution of a diagonally dominant periodic tridiagonal matrix system

$$A\mathbf{u}_{k+1} = \mathbf{d}_k, \qquad k = 0, 1, \ldots \tag{4.4}$$

where A is of the form (1.2), which is to be solved repeatedly in order to obtain the solution $\mathbf{u}_{k+1}$ on a new line $(k+1)\Delta t$ from the values of $\mathbf{u}_k$ on a previous line $k\Delta t$, for a given time increment Δt.

We consider specifically the following heat conduction equation

$$\frac{\partial U}{\partial t} - \frac{\partial^2 U}{\partial x^2} = 10(1-x)xt \tag{4.5}$$

with the initial condition

$$U(x, 0) = x(1-x), \qquad 0 \leqslant x \leqslant 1 \tag{4.6}$$

and the periodic condition specified in (4.3).

An analytical solution of (4.5) is given by[1],

$$U(x,t) = \frac{1+5t^2}{6} - \frac{5}{8}\sum_{m=1}^{\infty} \frac{\cos 2m\pi x}{m^6\pi^6}\{4m^2t - 1 + \exp(-4m^2\pi^2 t)\} - \sum_{m=1}^{\infty} \frac{\exp(-4m^2\pi^2 t)}{m^2\pi^2}\cos 2m\pi x. \tag{4.7}$$

Results are compared in Table 1 for the analytical solution given by (4.7) against the Crank–Nicolson scheme, for mesh sizes of 0.05 in the x and 0.005 in the t-directions, using the outlined periodic matrix solver algorithm in the solution process.

Table 1. Comparison of finite difference and analytical results at $t = 1.0$ and $x = 0.5$ for $\Delta x = 0.05$, $\Delta t = 0.005$

$t = 1.0$

	Crank–Nicolson finite difference solution	Analytical solution using eqn (4.7)
0.0	0.969709	0.097288
0.1	0.976694	0.979502
0.2	0.991224	0.993748
0.3	1.006442	1.000876
0.4	1.107494	1.019701
0.5	1.021499	1.023663

$x = 0.5$

t	Crank–Nicolson finite difference solution	Analytical solution using eqn (4.7)
0.1	0.178395	0.178753
0.3	0.247804	0.248318
0.5	0.385735	0.386512
0.7	0.590164	0.591573
0.9	0.861092	0.862901

REFERENCES

1. D. J. Evans, The numerical solution of the fourth boundary value problem for parabolic differential equations. *J. Inst. Maths. Appl.* 7, 61–75 (1971).
2. H. H. Ahlberg, E. N. Nilson and J. L. Walsh, *The Theory of Splines and their Applications*. Academic Press, New York (1967).
3. A. Bjorck and G. H. Golub, Eigenproblems for matrices associated with periodic boundary conditions. *SIAM Rev.* 19, 5–16 (1977).
4. R. S. Varga, *Matrix Iterative Analysis*. Prentice-Hall, Englewood Cliffs, New Jersey (1962).
5. H. S. Wall, *Analytical Theory of Continued Fractions*. Van Nostrand, Amsterdam (1948).